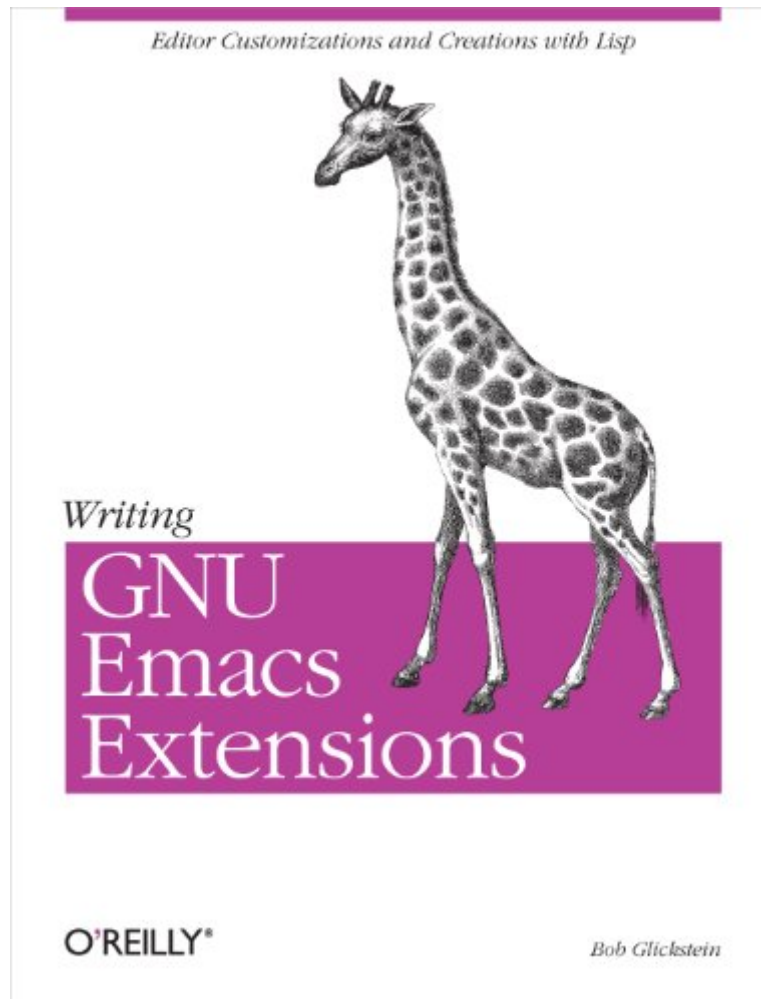


The book was found

Writing GNU Emacs Extensions: Editor Customizations And Creations With Lisp (Nutshell Handbooks)



Synopsis

Yes, it is possible to be all things to all people, if you're talking about the Emacs editor. As a user, you can make any kind of customization you want, from choosing the keystrokes that invoke your favorite commands to creating a whole new work environment that looks like nothing ever developed before. It's all in Emacs Lisp -- and in this short but fast-paced book. GNU Emacs is more than an editor; it's a programming environment, a communications package, and many other things. To provide such a broad range of functions, it offers a full version of the Lisp programming language -- something much more powerful than the little macro languages provided in other editors (including older versions of Emacs). GNU Emacs is a framework in which you can create whole new kinds of editors or just alter aspects of the many functions it already provides. In this book, Bob Glickstein delves deep into the features that permit far-reaching Emacs customizations. He teaches you the Lisp language and discusses Emacs topics (such as syntax tables and macro templates) in easy-to-digest portions. Examples progress in complexity from simple customizations to extensive major modes. You will learn how to write interactive commands, use hooks and advice, perform error recovery, manipulate windows, buffers, and keymaps, exploit and alter Emacs's main loop, and more. Each topic is explored through realistic examples and a series of successive refinements that illustrate not only the Emacs Lisp language, but the development process as well, making learning pleasant and natural.

Book Information

File Size: 560 KB

Print Length: 236 pages

Simultaneous Device Usage: Unlimited

Publisher: O'Reilly Media; 1 edition (April 1, 1997)

Publication Date: July 27, 2010

Sold by:Â Digital Services LLC

Language: English

ASIN: B0043M56SW

Text-to-Speech: Enabled

X-Ray: Not Enabled

Word Wise: Not Enabled

Lending: Not Enabled

Enhanced Typesetting: Not Enabled

Best Sellers Rank: #589,588 Paid in Kindle Store (See Top 100 Paid in Kindle Store) #64
inÂ Books > Computers & Technology > Programming > Languages & Tools > Lisp #157
inÂ Kindle Store > Kindle eBooks > Computers & Technology > Operating Systems > Unix #320
inÂ Books > Computers & Technology > Operating Systems > Unix

Customer Reviews

...is reading this book. Step 1 would be to read O'Reilly's "Learning Gnu Emacs" from cover to cover. Step 2 would be to start bookmarking "Info" pages in the Emacs and Elisp manuals (inside Emacs; Emacs can bookmark places in files you've edited, bookmark directories, bookmark Info pages, etc.); and then you are ready to read this book. While you can become proficient in Emacs just by learning a handful of commands, to be truly productive and happy you must learn most of the features and use them. This is a very long process (over a year for me, learning a little bit more each day). But what I've gained from the journey is invaluable. For example, one insight I've gotten is that Emacs can work very well for the novice (open/type/save/close) and the expert (write major mode to handle new language) equally well, and this idea can apply to any software project. (Sure, it sounds simplistic but the moment of "Aha!" is more profound than that.) This book is fairly small and progressively introduces new ideas in writing Lisp code to add functionality to Emacs. I think in retrospect the topics covered were well chosen because I have looked up the examples time and again to use code snippets. Step 4 in mastering Emacs is to read the newsgroup gnu.emacs.help every day for a few months, which will teach you about a great many features Emacs has that are not covered in any book (or covered very well, like term mode, font-lock and many more).

Glickstein offers practical solutions to gnu-emacs problems from the opening pages and only gets better from there. He introduces emacs-lisp topics gradually, and always in the context of solving a practical problem. One of the things I loved most about this book is that from the very first chapter, made emacs more usable by correcting some annoying traits that I had just accepted. Now I realize I can fix what I don't like! After finishing this book, a reader should be more confident in finding and modifying solutions contained at the gnu-emacs archive. Hopefully emacs's popularity will increase further as even more people take its destiny into their own hands. This wonderful introduction is a good start.

If you have taken the wise step and decided to learn emacs you're aware of the eLISP substructure underlying your C- and M- actions. Once you're aware that this power is there, you will invariably

want to use it to make some routine editing patterns faster / more efficient. I mean heck, you learned emacs to hack code in, didn't you? Why not hack emacs to make your hacking faster? In true geek fashion, I thought that this book would be, like so many of ORA's books, a canonical START on the monopoly board of computer / technology progress. It wasn't really. It started with introducing the notion of evaluating a lisp command string (in this case, making sure you have your ^H, ^? and Erase sorted out) - and goes from there. Too little time is spent on primitives (see, not really a programming guide as such) and instead uses a series of examples to make you think about how to use eLISP to handle an issue.....but that's not what you expect from an ORA book is it? You want the reference and the step-by-step -- you want to know you went to the source to get the answer and here was the path, right? Well for that you are actually better off going to gnu.org and reading the elisp manual there. It much more closely approximates the path that the ORA books (i.e. the camel book, etc.) take. Where this fits in -- a nice reference, maybe. IF YOU DO get this book, you'll find some handy examples and a few 'tricks of the trade. ' Nothing really great though.

This book is a collection of tutorials, it will help you start using and understanding Emacs. If you are a newbie, then it's a good starting point to use and customize Emacs. However, as a software engineer, I don't see the "big picture", meaning what really happens when Emacs starts, what are the main components of Emacs and how these components communicate with each other, which I think should be the first step needed to customize a software. So, this book has a gentle introduction to Emacs which is good if you just started using Emacs, but don't expect to be an expert in Emacs customization or have low level details of Emacs internals.

I found this book quite useful to get me started on writing Gnu emacs Lisp. It is clear and fairly well organized, and far easier to deal with than the standard documentation, which I find lacks the level of organization needed for so large a language. By "large" I mean mainly the number of functions available. Lisp as a language is not really hard to learn; it is just so different from many programming languages that it requires a few days of effort to get to "aha!". After that it becomes easy. The drawback to this book is that it doesn't take you far enough. It is an excellent start, and having worked through it you should be able to find your way around in the online or other emacs Lisp documentation. However, it lacks an index of emacs Lisp functions, or other similar reference material. I find this unfortunate, but it's not a show-stopper because once you get through this book you will know enough to use other reference material.

[Download to continue reading...](#)

Writing GNU Emacs Extensions: Editor Customizations and Creations with Lisp (Nutshell Handbooks) Learning GNU Emacs, Third Edition Land of Lisp: Learn to Program in Lisp, One Game at a Time! Controlar AutoCAD con AutoLISP/Visual LISP (Experto AutoCAD con Visual LISP nÂº 2) (Spanish Edition) Writing : Novel Writing Mastery, Proven And Simple Techniques To Outline-, Structure- And Write A Successful Novel ! - novel writing, writing fiction, writing skills - Stealing with Pride, Vol. 1: Advanced OSD Customizations for MDT 2013 and ConfigMgr 2012 R2 Harley Hahn's Emacs Field Guide Youdunit Whodunit!: How To Write Mystery, Thriller and Suspense Books (Writing Skills, Writing Fiction, Writing Instruction, Writing a Book) Microeconomic Theory: Basic Principles and Extensions Win32 Perl Programming: The Standard Extensions (2nd Edition) Eyebrow Extensions Manual: Learn Step by Step Procedures & How To Earn Money By Styling Eyebrows Eyelash Extensions Manual - Professional Student Manual Writing for Children (Writing Handbooks) Programming Embedded Systems: With C and GNU Development Tools, 2nd Edition The Subversive Copy Editor, Second Edition: Advice from Chicago (or, How to Negotiate Good Relationships with Your Writers, Your Colleagues, and ... Guides to Writing, Editing, and Publishing) The GNU Make Book Learning Debian GNU/Linux GNU/Linux Application Programming (Programming Series) Mystery Writing in a Nutshell: The World's Most Concise Guide to Mystery & Suspense Writing Stein On Writing: A Master Editor of Some of the Most Successful Writers of Our Century Shares His Craft Techniques and Strategies

[Dmca](#)